NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

REINFORCEMENT LEARNING APPLICATIONS TO COMBAT IDENTIFICATION

by

Emily M. Mooren

March 2017

Thesis Advisor:	Ying Zhao
Co-Advisor:	Walter Kendall

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704–0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** March 2017		**3. REPORT TYPE AND DATES COVERED** Master's thesis	
4. TITLE AND SUBTITLE REINFORCEMENT LEARNING APPLICATIONS TO COMBAT IDENTIFICATION				**5. FUNDING NUMBERS**
6. AUTHOR(S) Emily M. Mooren				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000				**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.				**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Crucial to the safe and effective operation of U.S. Navy vessels is the quick and accurate identification of aircraft in the vicinity. Modern technology and computer-aided decision-making tools provide an alternative to dated methods of combat identification. By utilizing the Soar Cognitive Architecture's reinforcement learning capabilities in conjunction with combat identification techniques, this thesis explores the potential for the collaboration of two. After developing a basic interface between Soar and combat identification methods, this thesis analyzes the overall correctness of the developed Soar agent to established truths in an effort to ascertain the level of system learning. While the scope of this initial research is limited, the results are favorable to a dramatic modernization of combat identification. In addition to establishing proof of concept, these findings can aid future research to develop a robust system that can mimic and/or aid the decision-making abilities of a human operator. While this research does focus on a sea-based, naval, application, the findings can also be expanded to DOD-wide implementations.

14. SUBJECT TERMS reinforcement learning, combat identification			**15. NUMBER OF PAGES** 75
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540–01-280-5500 Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

REINFORCEMENT LEARNING APPLICATIONS TO COMBAT IDENTIFICATION

Emily M. Mooren
Lieutenant Commander, United States Navy
B.A., University of Washington, 2005

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN NETWORK OPERATIONS AND TECHNOLOGY

from the

NAVAL POSTGRADUATE SCHOOL
March 2017

Approved by: Dr. Ying Zhao
Thesis Advisor

Walter Kendall
Co-Advisor

Dr. Dan Boger
Chair, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Crucial to the safe and effective operation of U.S. Navy vessels is the quick and accurate identification of aircraft in the vicinity. Modern technology and computer-aided decision-making tools provide an alternative to dated methods of combat identification. By utilizing the Soar Cognitive Architecture's reinforcement learning capabilities in conjunction with combat identification techniques, this thesis explores the potential for collaboration of the two. After developing a basic interface between Soar and combat identification methods, this thesis analyzes the overall correctness of the developed Soar agent to established truths in an effort to ascertain the level of system learning. While the scope of this initial research is limited, the results are favorable to a dramatic modernization of combat identification. In addition to establishing proof of concept, these findings can aid future research to develop a robust system that can mimic and/or aid the decision-making abilities of a human operator. While this research does focus on a sea-based, naval, application, the findings can also be expanded to DOD-wide implementations.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ADO	Air Defense Officer
CID	Combat Identification
COP	Common Operational Picture
CSG	Carrier Strike Group
DOD	Department of Defense
IFF	Interrogation Friend or Foe
LP	Learning Phase
MC	Mission Commander
NFO	Naval Flight Officer
OP	Operational Phase
POH	Probability of Hostility
PONH	Probability of Non-Hostility
RL	Reinforcement Learning
ROE	Rules of Engagement
TAO	Tactical Action Officer
TD	Temporal Difference
USN	United States Navy

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I'd like to thank a few people who were extremely important in the journey to completion. First, to my thesis advisors, Dr. Ying Zhao and Walter (Tony) Kendall, thanks for the patience, guidance and expertise. Also, a big thanks to Nate Derbinsky of the Wentworth Institute of Technology: your expertise in all things Soar was evident in all aspects. Thanks for the development of the Soar CID suite and the Windows user interface. Your knowledge of the system, its limitations and its benefits was needed and beyond welcome.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OBJECTIVE AND PURPOSE

The massive amount of information available to the tactical decision maker can overwhelm a single operator such as a Tactical Action Officer (TAO) or Mission Commander (MC). In an operational environment, the TAO or MC must identify and classify unknown aircraft quickly and correctly (Chief of Naval Operations [CNO], 2012). As the number of unknown aircraft increases, the corresponding amount of sensor data and decision-making information increases. By attempting to identify a program that will aid the TAO/MC's decision-making process, it may be possible to increase the effectiveness of the operator and, therefore, increase the safety inherent in the operational environment by reducing the amount of time that aircraft remain unclassified with respect to combat identification (CID). Through reinforcement learning (RL) solutions, the Soar: Cognitive Architecture could facilitate CID and, ultimately, mimic the cognitive process of a TAO/MC.

This thesis is a critical step in solving the problem of CID operator tasking overload that can be experienced by the TAO/MC decision maker, by identifying computer-aided decision-making tools that mimic the CID process through valid (accurate) RL. By evaluating the effects of RL on a simplified CID ruleset it is possible to evaluate the Soar Cognitive Architecture as a plausible framework to incorporate into TAO/MC duties. Ultimately, evaluating whether RL functions are a sufficient toolset to accurately mimic the cognitive functions of a TAO/MC in CID within a specific area of operations is crucial to proving the concept viable prior to extended research. Researching the potential benefits of RL could reframe the standard operating procedures of CID and the primary duties of the TAO/MC.

B. RESEARCH QUESTION

Evaluating a RL algorithm in conjunction with CID is a crucial step in research to ascertain feasibility of a cooperative system. Utilizing the SOAR Cognitive Architecture and a rudimentary CID matrix, this thesis will attempt to answer the following research question: "Does valid reinforcement learning of CID take place with SOAR cognitive architecture?"

Evaluation of the above research question will be achieved through the development and analysis of two results-oriented hypotheses.

- ***Hypothesis Ia***. *Incorporation of reinforcement learning/reward values into combat identification functions will decrease or not change the validity of the recommended action/identification provided.*
- ***Hypothesis Ib***. *Incorporation of reinforcement learning/reward values will increase the validity of the recommended action/identification provided.*

These hypotheses will be further discussed in Chapter IV

C. RESEARCH METHODOLOGY

Since no prior information for development of a CID decision-making matrix is available, the methods required to answer the proposed research question first require steps to develop the virtual environment and application. While limited past research has been done in this specific field, the principles of statistical analysis are still applicable to the data accumulated.

First, after a thorough examination of the information and knowledge of both fields (CID and RL), we will develop a rudimentary CID cognitive model of a TAO/MC. Taking into account inputs and methodology of CID itself, this will be done in such a manner that it can be easily translated into Soar application. The Soar CID agent developed will be tested against virtual track data in a limited simulation environment.

The data will then be collected in the simulation, first to establish a baseline for non-RL Soar CID, then to explore parameters of the Soar RL system. This exploitation of the parameters of RL in Soar will explore maximization of correctness in this application.

The overall correctness of the run compared to ground truth evaluation will be documented. The data will then be verified for statistical significance. Finally, based on the improvement or degradation in overall correctness in comparison to baseline sampling, we will be able to make assumptions of validity to the proposed employment.

D. POTENTIAL BENEFITS AND LIMITATIONS

The integration of CID and RL has the potential to overhaul the effectiveness CID as a process. By integrating a system that can adapt to local conditions for classification, the TAO/MC will have an additional tool to verify aircraft classifications, a safety net. As the efficiency of CID operationally increases this would have the two-fold benefits of freeing up the warfighter/operator for other tasking, and increasing the veracity of CID assumptions, thereby decreasing inaccurate identifications and decreasing completion time of the fix segment the “Kill Chain.”

Soar is a versitile RL program that can be adapted to suit many different disciplines. Integrating Soar and CID is a logical first step in the development of a CID system based on RL. Soar and the script written to mimic the cognitive functions of a TAO/MC are simplistic enough to test different variations of parameters and learning methods, policies that would be more difficult if done without. Automation of the RL implementation, even at this level, is streamlined.

The method in which the data and virtual track information have been inputted is labor intensive. While future research should integrate Soar and sensor outputs directly, removing the human operator from a portion of the process, the manual method of data entry to teach the RL limits the amount of data that can be entered and processed. In addition, there is currently no storage, or memory, for specific configurations or instances of tracks that it can build upon; each input is a new track.

Although the scope of this study is limited, partially due to its classification, the research is geared to set the stage for proving the feasibility of using artificial intelligence and learning programs in conjunction with CID. It is necessary to take the initial steps to prove the concept prior to advancing to more complicated scenarios. Through the testing of a basic model, establishing validity and lessons learned can and will help future research.

E. ORGANIZATION OF THESIS

In Chapter II, we will lay out current policies and background for both CID and RL, reviewing crucial terminology and ideas for both areas of study. Although previous research has been limited when combining the two, I will discuss possible implementation of RL and a cognitive architecture with respect to CID, and the possible methods of appropriate merging. Chapter II concludes with an explanation of the stated hypotheses. Chapter III develops the CID ruleset in an effort to mimic simplistic cognitive decision making of a TAO/MC and establishes parameters for the experimentation. Also, there is an introduction to the developed Soar CID application used to test the hypotheses. This chapter will also propose phases of learning appropriate to maximize RL return and accurate CID. Chapter IV is devoted to the statistical analysis of the results of the experimentation and analysis of the proposed hypotheses. Finally, Chapter V will summarize key points learned in the research and suggest further research possibilities that will allow the expansion of the ideas and concepts solidified throughout this thesis.

II. BACKGROUND

While the possible applications of reinforcement learning (RL) have extensively been studied in other domains, this has not included application to a combat identification (CID) process. This chapter will depict a baseline of knowledge within both RL and CID appropriate to the integration and experimentation. This chapter will also serve to cement the need for developing a new tool to aid the human decision maker in CID implementation.

A. COMBAT IDENTIFICATION

While the basic definition of CID holds true through multiple sources, The Under Secretary of Defense defines CID as "[c]apability to differentiate potential targets as friend, foe, or neutral in sufficient time, with high confidence, and at the requisite range to support weapons release and engagement decisions" (Department of Defense [DOD] and Joint Chiefs of Staff [JCS], 1996, p.II-4). It is a process critical to the safe and effective operation of warfighters through the Department of Defense (DOD). While all branches of the DOD participate in some form of CID, this research will focus on application to the United States Navy (USN) and its sea-based operators.

The objective of CID is primarily, "to correlate and assign a foe, friend or neutral identification label to a 'target'" (DOD and JCS, 1996, p. IV-C-1). The duties of CID in an operational USN environment primarily fall upon a few members of the carrier strike group (CSG) or independently deployed naval vessel. While the Air Defense Officer (ADO) is one of the ultimate decision makers in a CSG environment, on most vessels it is the Tactical Action Officer (TAO) who is tasked with the protection of the ship. A Mission Commander (MC) is a qualification assigned to the primary Naval Flight Officer (NFO) aboard an E-2 Hawkeye. In a CSG environment, a MC will aid the TAO and ADO in developing the Common Operational Picture (COP) by performing CID. All participants in creating a coherent COP operate off of common guidance and doctrine.

1. Why Is it Important?

It is imperative in modern battlespaces to know who is an enemy, who is a non-participant, and who is a friend (Joint Staff, 2014). This ability to classify surface vessels and aircraft in an environment is crucial to safe and effective combat and peacetime operations. CID done effectively can reduce the amount of possible friendly fire incidents (Joint Staff, 2014).

Most CID is just a part of a process to find, fix, track, target, engage, and assess (F2T2EA), commonly known as the "kill chain" (United States Air Force (USAF), 2014). The motivation to increase the accuracy and decrease the length of time for the "fix" segment of the "Kill Chain" is one of the most beneficial aspects of this CID application to aircraft identification.

2. Terminology

CID terminology and definitions hold weight and consequences. It is imperative to fleet operators that the lexicon of a TAO/MC is used with both the correct meaning and in the correct context. Defining the terminology of the process is a crucial step to understanding the cognitive structure of the warfighters tasked with the duty.

Contact: an instance of an aircraft which is represented on a local data system.

Track: an instance of an aircraft which is represented on a local data system, usually in conjunction with a datalink track number.

Target: an instance of an aircraft of interest.

Friend: "A positively identified friendly aircraft, ship or ground position" (HQ TRADOC, 2002).

Hostile: "A contact identified as an enemy upon which clearance to fire is authorized in accordance with theater rules of engagement" (HQ TRADOC, 2002).

Neutral: a contact identified neither as friend nor as foe.

3. Tools and Inputs

The sensor input to the decision maker can be divided into four categories: procedural, cooperative, non-cooperative methods, and intelligence ID fusion methods (Chief of Naval Operations [CNO], 2014). Procedural methods are based on the analysis of a target's motion or behaviors. While cooperative methods require the participation of the contact, non-cooperative methods will gather or extract information without any outside aid (CNO, 2014). Finally, methods based on the information obtained from intelligence networks. The ultimate identification could be based on information from all or some of the methods; the interpretation of the information provided is the primary task of the TAO with respect to CID.

Cooperative methods of CID are primarily useful in the identification of friendly and neutral aircraft. One of the most versatile and global is Identification, Friend or Foe (IFF). IFF is crucial to the safe and effective operation and identification of civilian and military aircraft across the world (DOD and JCS, 1996). The range of IFF systems and Modes are displayed in Table 1. While not all modes are used by all aircraft, there are combinations used by known entities that aid in identification. For instance, civilian aircraft are generally required to operate their transponder with Mode 3/A and Mode C active ("Transponder Requirements," 2006). Mode 1, 2, and 4 are primarily reserved for military aircraft (Department of the Navy [DON], 2013).

Table 1. IFF Systems Summary. Source: CNO (2014)

UNCLASSIFIED

IFF Systems (U)		
BASIC IFF MARK XII	IFF MARK XII(S)	IFF MARK XII(A)
Mode 1	Mode 1	Mode 1
Mode 2	Mode 2	Mode 2
Mode 3/A	Mode 3/A	Mode 3/A
Mode 4	Mode 4	Mode 4
SSR Mode C	SSR Mode C	SSR Mode C
I/P and Emergency modes	I/P and Emergency modes	I/P and Emergency modes
	Mode S	Mode 5 – secure mode, PIN
	Downlinked Air Parameters	LETHAL Mode

UNCLASSIFIED

Non-cooperative methods of data ingestion for CID include radar returns. For example, this data can be analyzed to localize the aircraft or for platform classification via jet engine modulation aspects (DOD and JCS, 1996).

There are multiple aspects of procedural control and this method of CID, such as point of origin or an aircraft operating on a predefined route in a predefined manner. An application of this behavior can be either minimum risk route (MRR) or return to force (RTF) profile (CNO, 2014)

While localizing a track or classifying its profile is not by itself a definitive identification of the hostility or friendliness of that contact, the profile can be used to help process the likelihood of either, or another classification (CNO, 2014). In addition, the particular responses to IFF transmissions need to be interpreted based on area rules of engagement (ROE) and guidance from regional commanders.

There is a wide range of inputs to the CID process, and all are a part of the overall picture to classifying the aircraft or contact. As information becomes available at any point in the Kill Chain that classification may or may not change based on the additional data (DOD and JCS, 1996). It is imperative that the operator or system tasked with CID is making decisions based on accurate and timely information.

4. Human Factors

While there are computer and weapons systems developed to aid the process of CID, the final decision making typically resides on the shoulders of the warfighter, TAO/MC, and the human elements of the process. Ultimately, the decision to interact with a target resides with the human decision maker. There have been instances of incorrect identification with devastating consequences. For example, the USS *Vincennes* incorrectly classified a commercial airliner as an Iranian F-14 on 3 July 1988 (Dottery, 1992). The decision was aided by the aegis weapons system recommendations and the time sensitivity of the matter, but the classification lead to the death of 290 civilians (Dottery, 1992).

The preponderance of current literature on human factors in CID centers around CID with respect to ground forces and combat in a land environment. Although the primary emphasis of this thesis revolves around naval implementation, there are lessons that are universal. There are human factors that influence CID decision making overall; stress, experience, personality, and expectations are the primary forerunners (Bryant, 2009). While this research does not focus on alleviating these factors, future research should focus on user interface and trust of the system to ensure that the computer decision aid is effective. If building a decision support aid, then human perception and differences in individuals need to be taken into account (Bryant, 2009).

B. COMPUTER AIDED DECISION-MAKING

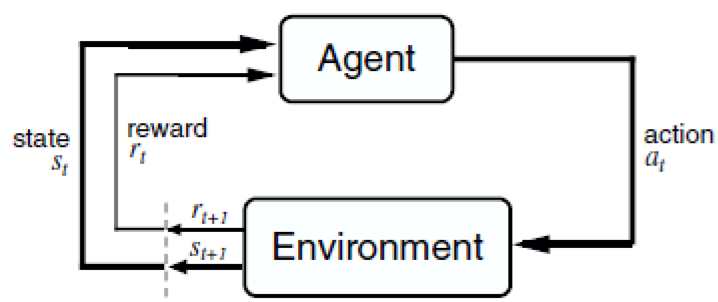
1. Reinforcement Learning

There are multiple methods of learning available to human and artificial systems in modern technology and human sciences. There are a few key factors that are of primary importance in reinforcement learning.

The basics of the interaction in RL take place between two components, the *agent,* and the *environment*. The *agent* is the component that learns and makes decisions and the *environment* is everything else, including the inputs to the agent for decision-making (Sutton & Barto, 1998). The agent's primary concern is to maximize rewards over time (Sutton & Barto, 1998).

In an application to CID, the agent would be the rules to classify *hostile* and *non-hostile* entities and reward values assigned to specified states. The choices that the agent makes depend on the preferences assigned to the track criteria at a given time. The environment consists of the observable space of a state and the human operator capable of rewarding the agent's action. As the operator rewards the agent's action (classification) the action is rewarded and the preference values are updated. The state consists of values assigned by sensors from the environment to a track at a specific time. In the loop depicted in Figure 1, once the possible reward values and state of a track are digested by the agent, an action is produced. In our implementation of RL CID, this action is a suggestion of identification classification awaiting user feedback.

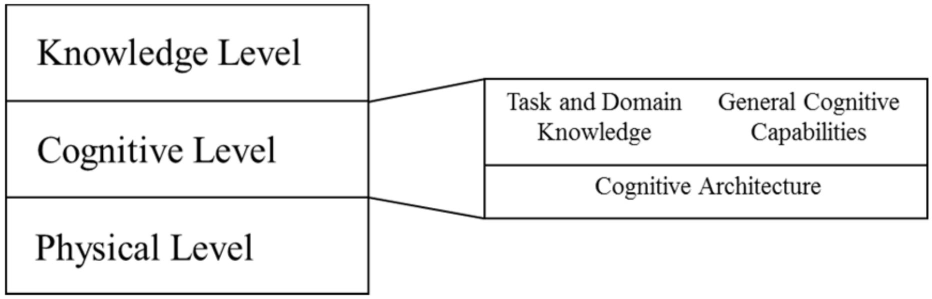
Figure 1. Agent-Environment Interaction. Source: Sutton and Barto (1998).



2. SOAR Cognitive Architecture

Building a structure that can translate operational knowledge to an encoded physical structure is the goal of Soar. As knowledge gets encoded into a system, the flexibility and adaptability of the system improves and exceeds the capabilities of systems lacking cognition (Laird, 2012). Figure 2 is a display of the intersection between Soar and the hierarchy of a physical/human decision maker.

Figure 2. Levels of Analysis of an Agent. Adapted from: Laird (2012)



Sitting above the physical level of signals and electrons, a cognitive architecture attempts to draw out the process knowledge and decision-making abilities of the human decision maker. "[A] cognitive architecture provides the fixed processes and memories and their associated algorithms and data structures to acquire, represent, and process knowledge about the environment and tasks for moment-to-moment reasoning problem solving and goal-oriented behavior" (Laird, 2012, p. 8). While this statement covers a multitude of possible applications, from chess to stacking blocks applications, the bottom line remains: the cognitive architecture presents an opportunity that could accurately be translated into a CID process.

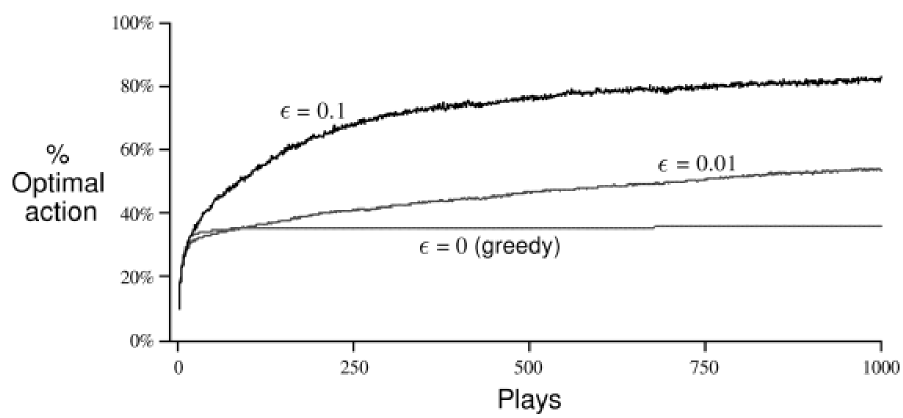
How the inputted data is treated is crucial to an effective RL system. Soar allows for the user to easily alter parameters of RL to suit their particular environment. While there are numerous parameters that can be changed to suit a RL application, the key components that will be explored in this thesis are learning-policy, exploration strategy and learning rate (Laird & Congdon, 2015).

There are two learning-policies available in Soar/RL: Q-Learning and SARSA. The two algorithms control how the data will be treated and how the expected future reward is chosen (Laird, 2012). Both are based on the concept of Temporal Difference (TD) learning, where specific methods estimate value functions prior to user input to modify the final reward (Eden, Knittel, & Uffelen, 2017). Q-learning is an Off-Policy TD method where the future reward is maximized and SARSA is a TD method where the future reward is the value of the selected operator (Laird, 2012).

Once the learning policy has been established, the important parameter decides how the actions will be chosen. As an agent can only improve when integrated with an environment, the environment needs to be explored. There are multiple exploration strategies in Soar. An exploration policy allows for decision making based on numeric preferences (Laird, 2012). There are two main methods: ε-greedy and softmax.

Greedy strategies look to exploit immediate maximized rewards (Sutton & Barto, 1998). The integration of ε adds a randomness to the selection. As ε decreases there is less randomness in selection; as it increases there is more. E-greedy strategies seek to maximize reward return, but may sometimes select an action at random. The utility of randomness has been proven in certain scenarios. The performance improvement overall with a higher degree of randomness, ε=0.1 in comparison to the other two depicted selections, is shown in Figure 3. The ε-greedy methods perform more optimally due to their continued exploration (Sutton & Barto, 1998). Without injecting randomness, the greedy strategy remained locked or stuck, selecting suboptimal actions.

Figure 3. E-greedy Performance Comparison.
Source: Sutton and Barto (1998)



A comparison of ε-greedy action-value methods. Data gathered from the application of a 10 armed bandit problem.

The second exploration strategy is softmax. Softmax behaves like greedy strategies in selecting the maximum reward but ranks and weighs the remaining actions depending on associated value estimates (Sutton & Barto, 1998). A variation of softmax is the Boltzmann distribution, which uses an additional variable called "temperature" to further affect the possibility of randomness. Temperature is a whole integer value, which is used to affect the ranking of the value estimates. As the temperature increases, all actions will become more equally probable. As the temperature decreases, actions will have greater difference in the probability of their selection, primarily based on the value estimates. A temperature setting of 0 will act much like a greedy strategy (Sutton & Barto, 1998). Soar sets a default temperature value of 25.

ε can be a parameter in each of the stated exploration strategies, its intention is to inject an amount of randomness into the agent. This could be beneficial to mimic the different environment and human applications.

Deciding which exploration strategy would be most useful is important because it will determine if an environment is still being explored or if it is being exploited. In terms of the two main strategies discussed earlier there may be benefits of one over the other based on variable settings. E-greedy is primarily an exploitation strategy, but as ε increases, there is more exploration due to the randomness. Softmax/Boltzmann is a combination determined by the temperature setting. The higher the temperature, the more exploration and the lower the temperature the system is biasing toward the best action, or maximum reward value, exploitation (Lewicki, 2007). Exploration versus exploitation has long been considered a dilemma (Lewicki, 2007): What is the appropriate amount of each? This will depend on the tasking of the RL application. In the context of CID, this has not been researched.

The selection of the learning rate is also important to developing a stable RL system. The default value for learning rate in Soar is 0.3, with a range of 0–1. If the learning rate is set approaching one, the system will learn quickly. If the learning rate is set approaching zero, the system will learn more slowly; when set at 0, the system will not update reward values (Eden et al., 2017). To stabilize a RL application it is feasible to lower the learning rate once the percentage of correct decisions has maximized. This could limit the impact of anomalous operator feedback issues but also negatively impact the system if the environment changes drastically. The constancy of the environment and the trust in the operators should have bearing on decisions to affect the learning rate in an operational implementation.

3. Cognitive Functions in CID

The translation of the cognitive functions of a TAO/MC in a CID context is not something that has been studied intensively. Although there are a few analyses of human decision making with respect to the discipline, there is not a definitive guide available at this level. Interpretations of previous research must be extrapolated to compare. One of the benefits of Soar Cognitive Architecture is that it assumes the bulk of the cognitive processes required to translate human to a machine. *The Cognitive Process of Decision Making* corroborates the cyclic tendencies of the decision making process and feedback loops to achieve a more accurate, satisfying, result (Wang & Ruhe, 2007). While there are methods of mapping CID decision making, the research focuses on the human parameters, and not necessarily on replicating the process in a machine (Bryant, 2009).

III. EXPERIMENTATION

Of primary importance to testing the hypotheses is developing an interface capable of accepting data entry and managing the algorithms of reinforcement learning (RL). The configuration of the Soar agent file is kept to a minimal amount of complexity at this stage of research in an effort to perform proof on the concept in this theater of study.

A. DEVELOPMENT OF CID RULESET

While Soar and RL have been proven in the past to excel at a variety of tasks the application to real-world scenarios demands a way of communicating with Soar. Pulling from the inputs to CID as described in Chapter II, we can extrapolate a few concepts that allow for a basic model of TAO/MC decision making.

CID is a process, with the classification of the track the end result. Since no one parameter leads to a full description of a track, the identification and subsequent classification of a track is a set of evaluations of the values for each parameter. In the course of interpreting a simplified CID process, we paired down the possible parameters to scope the project. While the factors that contribute to aircraft identification in a real-world environment are many, as was briefly discussed in Chapter II, the scope of this trial is limited to a four separate criteria: coordinates of the virtual track in a three-dimensional physical space (x, y, z), and one Interrogation Friend or Foe (IFF) value (Mode IV). The physical coordinates of the track represent a single point in time and mimic the profile of the contact based on procedural CID methodology.

Again, the resulting classification of a track is a combination of evaluations. While this could take the form of a series of “if > then” statements that allow an operator to achieve a classification based on the culmination, knowledge of Soar limitations due to the inputs to “state” requires a slightly different interpretation. There is not a method that allows for easy implementation of a complex compounding evaluation. Deconstructing the CID process to suit the Soar environment we make a few assumptions.

- Each parameter has an associated possibility of “hostility” or “non-hostility” based on its evaluation. This will be initially defined as probability of hostility (POH).

- The cumulative value of the POH can be used to ultimately evaluate the track.

- A range of POH values can be assigned to classifications of tracks (i.e., *hostile, non-hostile*).

Since variables are based on real-world parameters, the value set can be modified to suit specific geographical locations and political situations.

Application to CID takes the form of a set of logical rules. The values are not based on any real world scenario or parameters but a set of rules developed to test hypotheses in the scope of this thesis. The first set of “if > then” statements pulls from a procedural CID method.

- If the track has a determined location (x, y) ***less*** than (A, B) the POH assigned to that track is *n1*.

- If the track has a determined location (x, y) ***greater*** than (A, B) then the POH assigned to the track is *n2*.

- If the track has a determined altitude (z) ***less*** than C then the POH assigned to the track is *n3*.

- If the track has a determined altitude (z) ***greater*** than C then the POH assigned to the track is *n4*.

The following statements draw from cooperative CID methodology.

- If the IFF Mode 4 evaluation of the track is ***negative*** then the POH assigned to the track is *n5*.

- If the IFF Mode 4 evaluation of the track is ***positive*** then the POH assigned to the track is *n6*.

The initial value of *n* will have bearing on how quickly the Soar CID Application establishes a “learned” profile. RL totals the *n* value to arrive at a cumulative recommendation of POH. .

We will assign the following values to the A = 10, B = 10, C=6.

Therefore, an example of a track with the determined state (x=5, y=5, z=5, mode 4 = positive) would have an evaluation as follows:

$$POH = n1 + n3 + n6$$

The main goal of Soar RL is to maximize rewards over time. While there are environmental variables that can be modified, the RL program needs to be able to change the reward values to learn. The remaining variable that is a candidate for a reward value in the proposed ruleset is n.

While it is possible to logically assume that a lower cumulative POH would classify a track as less hostile, or possibly friendly, this does not work in RL. If there is not reward value assigned for classifying a track as *non-hostile* then there is no benefit for the system to choose that result. The system needs a balanced rule to reward the agent for choosing a non-hostile parameter. This will be known as probability of non-hostility (POHN). Therefore, each rule will have a *hostile-n* value (POH) and a *non-hostile-n* value (PONH). An example of the update to the previous "if > then" rules are:

- If the track has a determined location (x, y) ***less*** than (A, B) the POH assigned to that track is *n1*.
- If the track has a determined location (x, y) ***less*** than (A, B) the PONH assigned to that track is *n2*

If the Soar agent suggests *hostile* in the two example rules, and the Operator agrees with the agent, it is given feedback to change the *n* values to reflect the Operator preference. The change in *n1* and *n2* depends on the learning policy and exploration algorithm selected.

1. Basic Rules

This leads to translating the plain language rules into Soar CID Rules. Soar CID Rules are created using soar programming language and parameters as described by REFERENCE (Laird & Congdon, 2015). In this case, the rules were numbered to best track their usage. For example, Rule #1 has both a *hostile* and *non-hostile* variation with separate *n* (reward values). A specific example of the translation is depicted in Table 2. Values assigned to A, B, and C remain as stated previously.

Table 2. Plain Language to Soar Language of Rules

Plain Language	Soar Rule
If the track has a determined location (x, y) ***less*** than (A, B) the POH assigned to that track is *n1*.	sp {simple*eval*hostile*rule1 (state <s> ^name simple ^operator <op1> + ^io.input-link.features <f>) (<op1> ^name hostile) (<f> ^x < 10 ^y < 10) --> (<s> ^operator <op1> = 0.0001) }
If the track has a determined location (x, y) ***less*** than (A, B) the PONH assigned to that track is *n2*	sp {simple*eval*non-hostile*rule1 (state <s> ^name simple ^operator <op1> + ^io.input-link.features <f>) (<op1> ^name non-hostile) (<f> ^x < 10 ^y < 10) --> (<s> ^operator <op1> = 0.9999) }

Soar language for Rule #1 Hostile and Rule #1 Non-Hostile. POH(*n1)* for Rule #1 =0.0001. POHN(*n2)* for Rule #1=0.9999.

The full set of CID rules that will be used in this research and their assigned POH/PONH is shown in Table 3.

Table 3. CID Rules

Rule Name	Parameter	Starting POH/PONH Values
Rule 1 Hostile	x < 10 ; y < 10	0.0001
Rule 1 Non-Hostile	x < 10 ; y < 10	0.9999
Rule 2 Hostile	z < 6	0.2
Rule 2 Non-Hostile	z < 6	0.8
Rule 3 Hostile	Mode 4	0.0001
Rule 3 Non-Hostile	Mode 4	0.9999
Rule 4 Hostile	x >10 ; y > 10	0.0001
Rule 4 Non-Hostile	x >10 ; y > 10	0.9999
Rule 5 Hostile	z > 6	0.2
Rule 5 Non-Hostile	z > 6	0.8

Rule #1 and Rule #4 are complementary, as are Rule #2 and Rule #5. Each rule has a Hostile and Non-Hostile variant with a corresponding reward value (POH/PONH). Rule #3 does not have a paired rule for non-Mode 4 parameters.

Due to the nature of the rules, the rules are either "tripped" or not. If a track meets a rule's condition, then the rule is "tripped" and assigned associated reward value/POH. The possible combinations of "tripped" and "non-tripped" rules sum up to eight separate track variations. In an effort to create a stable or ground truth about each of the tracks, an assignment of *hostile* or *non-hostile* has been assigned to each of the variations of tracks. This is in an effort to judge the veracity of the Soar/RL result as it learns against ground-truth values. The ground-truth values and parameters of each track are given in Table 4. While no specific significance is placed on 12 or 5, its intention is to trip above 10 or below 6 based on Rules #1/4 and Rule #2/5, respectively.

Table 4. Ground Truth Values of Tracks

Track #	X-value	Y-value	Z-value	MODE	Hostility
1	5	5	5	0	Y
2	12	12	5	0	Y
3	5	5	12	0	Y
4	5	5	5	4	N
5	5	5	12	4	N
6	12	12	12	4	N
7	12	12	5	4	N
8	12	12	12	0	N

For the purposes of the experiment, the truthful "hostility" is annotated. This ensures that the feedback is given when "training" the system is uniform and expected. "Y" means *hostile* and "N" means *non-hostile.*

Since the sample size, the pool of possible track configurations, is extremely limited based on the scoped parameters, the repetition of tracks 1–8 is unavoidable. Data entry and track sampling will occur in two manners. The first is through an ordered, equal ratio of tracks 1–8. The second is a randomized sampling of tracks 1–8. This is done to compare the different environments and evaluate the results.

2. Reward Value-Functions

Reward values may factor dramatically in the RL veracity in a common operating environment. At the beginning reward values, *n,* are set at a default value and then those values will change based on the "training" given to the RL system to reflect the operating environment and specifics of the theater. While the starting reward values assigned to a rule can be modified to suit the weight and consequence of the parameter, the starting value assigned to each rule in the experimentation has no correlation to real-world parameters.

B. SOAR SETTINGS

While there are a variety of different settings than can affect RL in the Soar environment, the experiment will first focus on default policies and rates. We then delve into different variations of the parameters to maximize correctness.

The learning-policy selected for the bulk of the basic testing is SARSA. The initial learning rate is set at default, 0.3. This allows for a moderately fast training phase. Iterations of the parameters also explore a decreased learning rate in the latter stages of application to minimize the swing of reward values. The default exploration policy is softmax. E-greedy and boltzmann strategies will be explored and compared.

Also, a sample testing will be generated in an effort to understand and demonstrate the immediate differences between the tested parameters. This sample will be one iteration of tracks 1–8, ordered, utilizing separate learning methods and exploration policies. The results will note the change in the reward value between different sets of parameters.

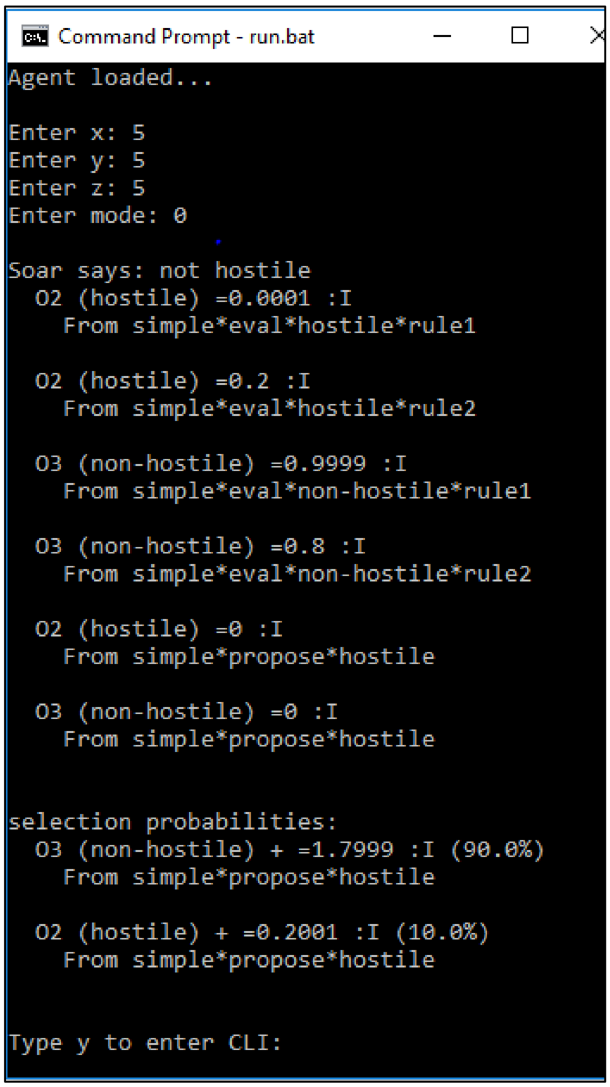
C. SOAR CID APPLICATION

While the Soar software suite is comprised of a set of files that are all required to work in concert, there are a few dynamic selections that will be addressed. The components of the agent folder are the rules created to support the environment.

The Soar Cognitive Architecture has been adapted to tie into an input mechanism utilizing the Windows Command Prompt. A small amount of programming allows for the

Soar RL functions to be manipulated and controlled through an easier interface. This allows for relatively easy, albeit labor intensive, entry of the virtual track parameters (Table 4) into the Soar CID program. Although this is not realistic for shipboard usage or a larger sample size, this is sufficient for the scope of this thesis. An example of entry or Track 1 into an untrained system is shown is Figure 4. Once the Soar CID agent is loaded, the operator is prompted to enter track parameter values.

Figure 4. Soar CID—Windows Integration



```
Command Prompt - run.bat
Agent loaded...

Enter x: 5
Enter y: 5
Enter z: 5
Enter mode: 0

Soar says: not hostile
  O2 (hostile) =0.0001 :I
    From simple*eval*hostile*rule1

  O2 (hostile) =0.2 :I
    From simple*eval*hostile*rule2

  O3 (non-hostile) =0.9999 :I
    From simple*eval*non-hostile*rule1

  O3 (non-hostile) =0.8 :I
    From simple*eval*non-hostile*rule2

  O2 (hostile) =0 :I
    From simple*propose*hostile

  O3 (non-hostile) =0 :I
    From simple*propose*hostile


selection probabilities:
  O3 (non-hostile) + =1.7999 :I (90.0%)
    From simple*propose*hostile

  O2 (hostile) + =0.2001 :I (10.0%)
    From simple*propose*hostile


Type y to enter CLI:
```

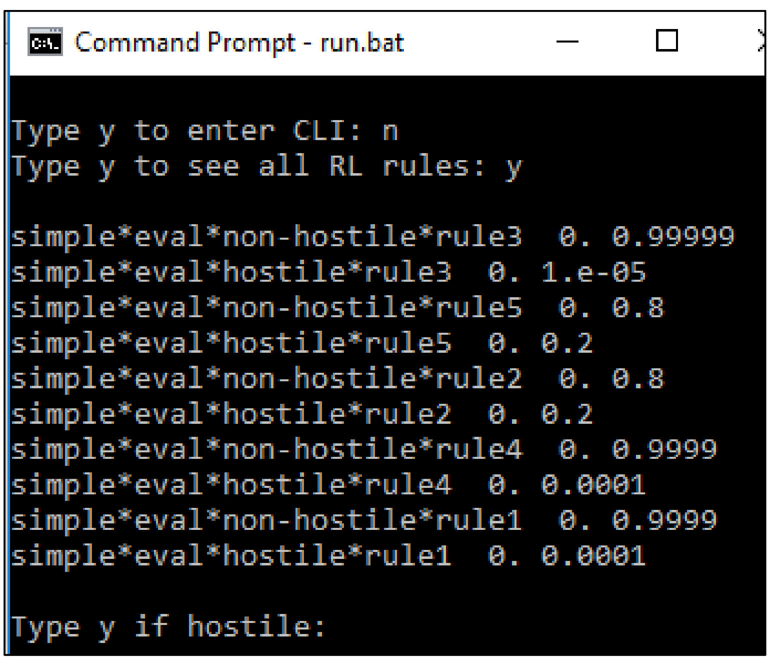
Example entry for Soar CID track entry. Operator separately entered x, y, z and mode parameters. The initial recommendation of Soar CID is displayed. Operator feedback has not been entered.

Soar recommends a classification: "Soar says: not hostile." Below the recommendation are the rules that were "tripped" with specific track conditions: Rule #1 (hostile and non-hostile), Rule #2 (hostile and non-hostile). The associated reward values are tallied

The Soar CID program has been configured to display the percentage of probability of non-hostility and hostility based on the current reward values that in its memory. In the instance above, PONH = 1.7999 and POH = 0.2001. In the above case, Track 1 has a 90% probability of being "non-hostile" and a 10% probability of being "hostile." This is a translation of the total POH and PONH beside it. The total POH + PONH is 2.0, 1.7999 / 2.0 = .89995 or 90%.

The Operator next has the opportunity to view all RL rules and their associated reward value before proceeding to the feedback stage. The remainder of the reward values in current memory is shown in Figure 5.

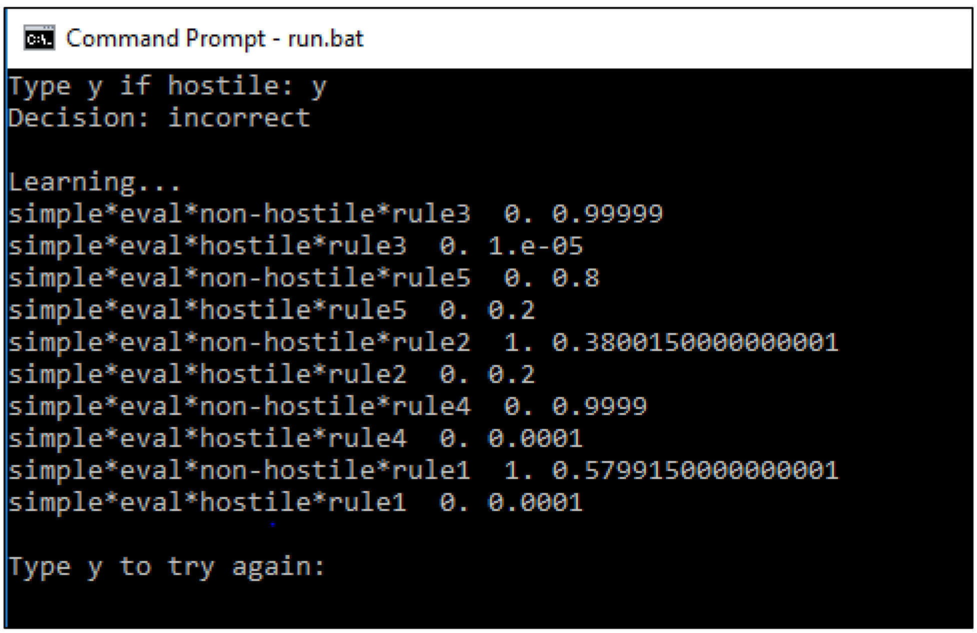
Figure 5. Operator Selection of All RL Rules and Current Values.



If Operator enters "y" at the prompt then all non "tripped" rules and current values will be displayed.

The final input for each track will be operator feedback. In this initial configuration of the Soar CID application if the operator presses “y,” then is to confirm that the Track entered is evaluated as “hostile.” If the track is “non-hostile” then the operator would enter a non “y” value. The feedback stage of Track 1 evaluation is shown in Figure 6. Since the initial recommendation of Track 1 was “non-hostile” (Figure 4) and the operator entered “y” for a “hostile” evaluation, the “decision” line depicted in Figure 6 states “incorrect.” In this instance, Soar and the operator did not agree on the classification of the track.

Figure 6. Learning Mode of the Soar CID Application



Operator feedback of “y” for hostile leads Soar to evaluate its reward valuations and adjust for future attempts.

Soar CID will then apply the operator feedback in the form of modifying the reward values to improve future evaluations. Since Rule #1 and Rule #3 were “tripped” those are the reward values that are modified. Until the correct rules are accepted, maximizing rewards, the reward value assigned to the incorrect selection will degrade. In an instance of the Soar recommendation being “correct,” the reward values will increase. The specific calculation of reward value alteration is based on specific Soar parameters (learning policy, exploration policy, learning rate). The operator now has the opportunity to test another track with the new “learned” reward values.

In this initial version of Soar CID, there is no stored memory to build upon outside of one initialization in the Soar CID program. If the operator does not select, “try again” then the next time the program runs it will again be the “untrained” system. This has ramifications for the potential sample size due to operator mistakes.

D. VARIATIONS AND SAMPLING

Once the data from the Soar CID application has been accumulated it will be exported to Excel for summarization and analysis. Since there is no stored memory between each continuous assessment, one assessment will be referred to as a “run.” Each run will be a sampling of Tracks 1–8 (Table 4) in either sequential or random order, in various recurrences.

As the hypotheses are based on the comparison and proving that the system improves, “learns,” we must first establish a baseline. The baseline will be established by allowing the application to run without learning. Each track will be evaluated by Soar CID without any feedback from the operator. The percentage of correctness based on the ground truth evaluations listed in Table 4 and will be established as our base value.

Further iterations will be concerned with establishing if the system can improve or “learn” and modifying Soar CID RL parameters to maximize the overall correctness. This will be done based on the principles explored in Chapter II and in previous research. Balancing exploration and exploitation is crucial to developing an adaptable system (Tokic, 2010; Sutton & Barto, 1998). Therefore, the modification of exploration strategies and learning rates will help to establish the best parameters for Soar CID. Comparison analysis of the baseline numbers and the other variations will potentially show better parameter settings for this application. The samples will also be evaluated for statistical significance in comparison to the baseline numbers and each other.

E. PHASES OF REINFORCEMENT LEARNING APPLICATION

In addition to data analysis of random and ordered samples, the concept of a teaching a system prior to placing it in operation will be evaluated. With the utilization of default values, conceivably, the initial stages of learning will produce less correct results than the latter stages; the system will learn.

While the initial teaching of the RL system is potentially crucial to establishing a higher overall correctness, it is possible to export the "taught" system and establish a basic Soar CID agent where further usage will mean greater overall correctness and fewer inaccurate recommendations. The comparisons between the latter taught models will help to further evaluate the validity of the hypotheses. We propose two phases to Soar CID implementation.

1. Learning Phase

Numerous runs will be completed to assess when the Soar CID agent achieves a relatively stable state, the learning phase (LP). Due to the small sample size of the track pool, it is not expected to result in 100% overall correctness. The Soar CID agent file will then be exported for use during multiple iterations of the follow-on phase. Since the current Soar CID application has no in program memory, this is crucial due to the instability of the virtual environment. The only way to build upon the current learning is either to make no mistakes or to export and modify an additional Soar CID application with new values.

2. Operational Phase

After the LP, we propose an operational phase (OP). While the main idea behind OP is that the overall correctness metric is not influenced by the LPs inherently low accuracy, there are beneficial considerations that can be explored.

Parameters of RL can be modified such that an incorrect entry during the feedback stage or a unique set of track parameters does not dramatically affect the reward values. During the OP both the exploration strategy and learning rate be modified to evaluate the effect on the overall results.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DATA ANALYSIS

A. RESULTS

1. Baseline Results—No Learning

The track pool analyzed without any reinforcement learning (RL) applications are stated in Table 5. The overall correctness of a non-learning application of the tracks is five correct out of eight, 62.5%. The percentage of correctness without learning is established as a baseline for comparison to further runs and parameter testing. Extrapolated to a sample size of 48 tracks, this creates a ratio of 30 out of 48 correct, in a sequential sampling of the track pool.

Table 5. Baseline Run - No Learning

TRACK #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE (Y/N)	CORRECT	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	OVERALL CORRECTNESS
1	Soar says: not hostile	Y	N	90.0%	10.0%	62.50%
2	Soar says: not hostile	Y	N	90.0%	10.0%	
3	Soar says: not hostile	Y	N	90.0%	10.0%	
4	Soar says: not hostile	N	Y	93.3%	6.7%	
5	Soar says: not hostile	N	Y	93.3%	6.7%	
6	Soar says: not hostile	N	Y	93.3%	6.7%	
7	Soar says: not hostile	N	Y	93.3%	6.7%	
8	Soar says: not hostile	N	Y	90.0%	10.0%	

The results from a Soar CID run where no RL was applied.

2. Sequential v Random Sampling with Default Parameters

The sequential sampling resulted in an overall correctness of 72.91%. The results for a run of 48 tracks are shown in Table 6, 1–8 repeating. The RL parameters are set to the default rates discussed in Chapter III, they include: softmax, ε 0.1, learning-rate 0.3

Table 6. Run 1: Sequential Sampling, Default Parameters

Track #	T+	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	0	Soar says: not hostile	Y	N	90.0%	10.0%	72.92%
2	0	Soar says: not hostile	Y	N	87.3%	12.7%	
3	0	Soar says: not hostile	Y	N	87.3%	12.7%	
4	0	Soar says: not hostile	N	Y	86.2%	13.8%	
5	0	Soar says: not hostile	N	Y	89.0%	11.0%	
6	0	Soar says: not hostile	N	Y	90.6%	9.4%	
7	0	Soar says: not hostile	N	Y	87.2%	12.8%	
8	0	Soar says: not hostile	N	Y	80.0%	20.0%	
1	1	Soar says: hostile	Y	Y	33.0%	67.0%	
2	1	Soar says: hostile	Y	Y	61.2%	38.8%	
3	1	Soar says: not hostile	N	N	58.7%	41.3%	
4	1	Soar says: hostile	N	N	55.0%	45.0%	
5	1	Soar says: not hostile	N	Y	98.6%	1.4%	
6	1	Soar says: not hostile	N	Y	90.8%	9.2%	
7	1	Soar says: not hostile	N	Y	84.9%	15.1%	
8	1	Soar says: not hostile	N	Y	64.4%	35.6%	
1	2	Soar says: hostile	Y	Y	0.0%	100.0%	
2	2	Soar says: hostile	Y	Y	49.9%	50.1%	
3	2	Soar says: hostile	Y	Y	22.0%	78.0%	
4	2	Soar says: hostile	N	N	55.4%	44.6%	
5	2	Soar says: not hostile	N	Y	94.9%	5.1%	
6	2	Soar says: not hostile	N	Y	88.8%	11.2%	
7	2	Soar says: not hostile	N	Y	86.0%	14.0%	
8	2	Soar says: not hostile	N	Y	55.0%	45.0%	
1	3	Soar says: hostile	Y	Y	0.0%	100.0%	
2	3	Soar says: not hostile	Y	N	44.4%	55.6%	
3	3	Soar says: hostile	Y	Y	22.8%	77.2%	
4	3	Soar says: hostile	N	N	52.3%	47.7%	
5	3	Soar says: not hostile	N	Y	95.1%	4.9%	
6	3	Soar says: not hostile	N	Y	90.1%	9.9%	
7	3	Soar says: not hostile	N	Y	97.4%	2.6%	
8	3	Soar says: hostile	N	N	46.7%	53.3%	
1	4	Soar says: hostile	Y	Y	0.0%	100.0%	
2	4	Soar says: not hostile	Y	N	17.5%	82.5%	
3	4	Soar says: not hostile	Y	N	29.8%	70.2%	

Track #	T+	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
4	4	Soar says: hostile	N	N	45.4%	54.6%	
5	4	Soar says: not hostile	N	Y	100.0%	0.0%	
6	4	Soar says: not hostile	N	Y	100.0%	0.0%	
7	4	Soar says: not hostile	N	Y	100.0%	0.0%	
8	4	Soar says: not hostile	N	Y	72.9%	27.1%	
1	5	Soar says: hostile	Y	Y	0.0%	100.0%	
2	5	Soar says: hostile	Y	Y	0.0%	100.0%	
3	5	Soar says: hostile	Y	Y	2.4%	97.6%	
4	5	Soar says: not hostile	N	Y	60.7%	39.3%	
5	5	Soar says: not hostile	N	Y	100.0%	0.0%	
6	5	Soar says: not hostile	N	Y	100.0%	0.0%	
7	5	Soar says: not hostile	N	Y	100.0%	0.0%	
8	5	Soar says: hostile	N	N	58.5%	41.5%	

The randomly ordered sampling resulted in an overall correctness of 77.08% which is displayed in Table 7.

Table 7. Run 2: Randomized Sampling, Default Parameters

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE (Y/N)	Correct ?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
2	Soar says: not hostile	Y	N	90.0%	10.0%	77.08%
3	Soar says: not hostile	Y	N	87.3%	12.7%	
4	Soar says: not hostile	Y	N	87.3%	12.7%	
5	Soar says: not hostile	N	Y	86.2%	13.8%	
6	Soar says: hostile	N	N	89.0%	11.0%	
7	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	94.6%	5.4%	
8	Soar says: not hostile	N	Y	91.2%	8.8%	
8	Soar says: not hostile	N	Y	91.7%	8.3%	
1	Soar says: not hostile	Y	Y	92.0%	8.0%	
4	Soar says: hostile	N	Y	66.1%	33.9%	
5	Soar says: not hostile	N	Y	80.7%	19.3%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: not hostile	Y	N	30.9%	69.1%	

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE (Y/N)	Correct ?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
3	Soar says: not hostile	Y	N	100.0%	0.0%	
8	Soar says: not hostile	N	Y	88.0%	12.0%	
2	Soar says: hostile	Y	Y	48.7%	51.3%	
2	Soar says: hostile	Y	Y	37.5%	62.5%	
8	Soar says: not hostile	N	Y	74.1%	25.9%	
2	Soar says: hostile	Y	Y	35.1%	64.9%	
7	Soar says: not hostile	N	Y	63.1%	36.9%	
6	Soar says: not hostile	N	Y	89.6%	10.4%	
6	Soar says: not hostile	N	Y	88.5%	11.5%	
8	Soar says: hostile	N	N	67.1%	32.9%	
2	Soar says: hostile	Y	Y	30.3%	69.7%	
2	Soar says: hostile	Y	Y	26.3%	73.7%	
8	Soar says: not hostile	N	Y	97.5%	2.5%	
2	Soar says: hostile	Y	Y	28.1%	71.9%	
7	Soar says: not hostile	N	Y	56.7%	43.3%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
4	Soar says: hostile	N	N	100.0%	0.0%	
4	Soar says: not hostile	N	Y	100.0%	0.0%	
7	Soar says: hostile	N	N	66.8%	32.2%	
2	Soar says: hostile	Y	Y	36.4%	63.6%	
3	Soar says: hostile	Y	N	50.0%	50.0%	
8	Soar says: hostile	N	Y	74.2%	25.8%	
2	Soar says: hostile	Y	Y	36.8%	63.2%	
8	Soar says: not hostile	N	Y	0.0%	100.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
4	Soar says: hostile	N	N	20.0%	80.0%	
4	Soar says: not hostile	N	Y	100.0%	0.0%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
2	Soar says: not hostile	Y	Y	41.7%	58.3%	

Both sequential and randomized samples of the similar sample size exceed the baseline, non-learning proportion of overall correctness. The average improvement in overall correctness is 12.5%.

1. Statement of Hypothesis

In order to ultimately answer the research question stated in Chapter I, the problem will be analyzed by a hypothesis based on the central idea of RL: reward values. As the reward values continue to change through the operator/agent relationship and training, does this affect the overall accuracy of the Soar decision? Basically, does the system learn?

From that research question, we are proposing a hypothesis for analysis. In an attempt to establish proof of concept the hypothesis will concentrate on whether the outcome is affected. If the system displays a capacity to deliver increasing overall correctness, the system will have "learned." To accept that the system "learned," we must first consider that the incorporation of RL and CID was not successful (i.e., our null hypothesis).

a. Hypotheses Ho

Incorporation of reinforcement learning/reward values into combat identification functions will decrease or not change the validity of the recommended action/identification provided.

Therefore if the overall correctness of CID problems is increased by the incorporation of RL and associated reward values the alternative would be the following statement.

b. Hypothesis Ha

Incorporation of reinforcement learning/reward values will increase the validity of the recommended action/identification provided.

The system will have learned if the data can be proven to be significant with 95% certainty. With an established alpha value of 0.05 the corresponding probability (p-value) will need to be less than or equal to alpha.

As discussed, the data sample size is small but the corresponding probability of the non-learning baseline to Run 1 and Run 2 is $p=0.1375$ and $p=0.0599$, respectively. The p-values were calculated via the statistical proportions tools on vassarstats.net. In the initial testing, both p-values for Run 1 and Run 2 fail the established acceptable threshold.

4. Learning Phase Results

After multiple iterations of ordered sampling, the optimal combination resulted from an ordered sampling of four sets of tracks, totaling 32 total samples. Although the overall correctness is less than the results depicted in Run 2 (Table 7), the resulting reward values allowed for greater overall correctness in subsequent runs. The learning phase (LP) results are stated in Table 8. Sampled in segments of eight, the results fluctuate but eventually stabilize.

Table 8. Run 3. Learning Phase Results

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE? (Y/N)	CORRECT?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	Soar says: not hostile	Y	N	90.0%	10.0%	65.63%
2	Soar says: not hostile	Y	N	87.3%	12.7%	
3	Soar says: not hostile	Y	N	87.3%	12.7%	
4	Soar says: hostile	N	N	86.2%	13.8%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
6	Soar says: not hostile	N	Y	96.1%	3.9%	
7	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	79.7%	20.3%	
1	Soar says: not hostile	Y	N	100.0%	0.0%	
2	Soar says: not hostile	Y	N	81.4%	18.6%	
3	Soar says: not hostile	Y	N	78.8%	21.2%	
4	Soar says: not hostile	N	Y	100.0%	0.0%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE? (Y/N)	CORRECT?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
6	Soar says: not hostile	N	Y	94.4%	5.6%	
7	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	67.3%	32.7%	
1	Soar says: hostile	Y	Y	50.0%	50.0%	
2	Soar says: hostile	Y	Y	31.6%	68.4%	
3	Soar says: hostile	Y	Y	22.5%	77.5%	
4	Soar says: not hostile	N	Y	54.7%	45.3%	
5	Soar says: not hostile	N	Y	75.2%	24.8%	
6	Soar says: not hostile	N	Y	82.9%	17.1%	
7	Soar says: not hostile	N	Y	74.0%	26.0%	
8	Soar says: hostile	N	N	53.0%	47.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
2	Soar says: not hostile	Y	N	26.3%	73.7%	
3	Soar says: hostile	Y	Y	16.1%	83.9%	
4	Soar says: not hostile	N	Y	38.7%	61.3%	
5	Soar says: hostile	N	N	72.1%	27.9%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
7	Soar says: hostile	N	N	91.8%	8.2%	
8	Soar says: not hostile	N	Y	100.0%	0.0%	

5. Operational Phase Results

The first attempt at maximization of the operational phase (OP) utilized the default parameters as discussed in Chapter III. The results show a marked improvement over the base correctness of 62.5% as shown in Table 9. Once the LP was loaded, the OP operated on the rewards values produced from Table 8.

Table 9. Run 4: Operational Phase, Random Ordering, Default

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE? (Y/N)	CORRECT?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	Soar says: hostile	Y	Y	0.0%	100.0%	91.9%
2	Soar says: hostile	Y	Y	0.0%	100.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE? (Y/N)	CORRECT?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
4	Soar says: not hostile	N	Y	55.8%	44.2%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
3	Soar says: not hostile	Y	N	37.3%	62.7%	
8	Soar says: not hostile	N	Y	100.0%	0.0%	
3	Soar says: hostile	Y	Y	0.0%	100.0%	
8	Soar says: not hostile	N	Y	85.2%	14.8%	
5	Soar says: not hostile	N	Y	85.6%	14.4%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
5	Soar says: not hostile	N	Y	84.5%	15.5%	
8	Soar says: not hostile	N	Y	88.9%	11.1%	
5	Soar says: not hostile	N	Y	85.4%	14.6%	
5	Soar says: not hostile	N	Y	85.3%	14.7%	
2	Soar says: not hostile	Y	N	29.9%	70.1%	
5	Soar says: not hostile	N	Y	85.2%	14.8%	
4	Soar says: hostile	N	N	34.5%	65.5%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	
7	Soar says: not hostile	N	Y	100.0%	0.0%	
4	Soar says: not hostile	N	Y	64.3%	35.7%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	75.4%	24.6%	
2	Soar says: hostile	Y	Y	3.3%	96.7%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
4	Soar says: not hostile	N	Y	58.8%	41.2%	
7	Soar says: not hostile	N	Y	86.9%	13.1%	
2	Soar says: hostile	Y	Y	7.1%	92.9%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
4	Soar says: not hostile	N	Y	47.3%	52.7%	
7	Soar says: not hostile	N	Y	71.8%	28.2%	
4	Soar says: hostile	N	N	60.3%	39.7%	
8	Soar says: not hostile	N	Y	54.4%	45.6%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	

Track #	SOAR RECOMMENDATION	GROUND TRUTH HOSTILE? (Y/N)	CORRECT?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	Soar says: hostile	Y	Y	0.0%	100.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
4	Soar says: not hostile	N	Y	69.1%	30.9%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	51.7%	48.3%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
4	Soar says: not hostile	N	Y	67.0%	33.0%	
4	Soar says: not hostile	N	Y	71.2%	28.8%	
1	Soar says: hostile	Y	Y	0.0%	100.0%	
4	Soar says: not hostile	N	Y	72.5%	27.5%	
2	Soar says: hostile	Y	Y	0.0%	100.0%	
8	Soar says: not hostile	N	Y	54.0%	46.0%	
3	Soar says: hostile	Y	Y	50.7%	49.3%	
3	Soar says: not hostile	Y	N	41.7%	58.3%	
4	Soar says: not hostile	N	Y	61.9%	38.1%	
2	Soar says: hostile	Y	Y	9.0%	91.0%	
4	Soar says: not hostile	N	Y	65.2%	34.8%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: not hostile	N	Y	39.2%	60.8%	
2	Soar says: hostile	Y	Y	15.0%	85.0%	
5	Soar says: not hostile	N	Y	100.0%	0.0%	
2	Soar says: hostile	Y	Y	15.0%	85.0%	
8	Soar says: not hostile	N	Y	50.4%	49.6%	
6	Soar says: not hostile	N	Y	100.0%	0.0%	
8	Soar says: hostile	N	N	51.0%	49.0%	
8	Soar says: not hostile	N	Y	90.7%	9.3%	
7	Soar says: not hostile	N	Y	92.2%	7.8%	
4	Soar says: not hostile	N	Y	61.3%	38.7%	
3	Soar says: hostile	Y	Y	35.0%	65.0%	

The next OP testing used the same LP phase but changed the ε value to 0.05. Although the sample as still randomized, the overall correctness remained relatively stable but decreased slightly from 91.9% to 88.9%. A comparison of OP variations in multiple exploration strategies and parameters is featured in Table 10. While Runs 4, 5, and 6 are similarly high in the overall correctness metric, Run 7 falls short of even the untrained baseline, 48.0% to 62.5%.

Table 10. Operational Phase Parameter Exploration

Learning Policy	Exploration Strategy	Epsilon	Learning Rate	Overall Correctness	Sample Size
SARSA	SOFTMAX	0.1	0.3	91.89%	74
SARSA	SOFTMAX	0.05	0.3	88.89%	99
SARSA	GREEDY	0.1	0.3	91.00%	100
SARSA	BOLTZMANN	0.1	0.3	48.00%	100

A comparison of the exploration strategies applied in the OP. Run 5 (Appendix A) Run 6 (Appendix B) Run 7 (Appendix C)

6. Anomalies and Unexpected Results

While the preponderance of track iteration evaluations yielded results that logically paired with their POH/PONH and percentage, there were a few iterations in which Soar recommended the alternative classification, against obvious rewards. An example is Line 31 (Appendix A.) The percentage of PONH (53.9%) is higher than the POH (46.1%), but Soar recommended *hostile*. The ground-truth of this track is *non-hostile*. The system deliberately went contrary to the maximized reward and percentage. This occurred a few times in each Run, the percentage between the two, POH/PONH, is relatively close, in the 10% range overall. This has a direct correlation to the ε value chosen for the implementation. The system will continue to explore its environment with an element of randomness. While utilizing an ε value greater than 0, there will always be a number of agent-recommended decisions that are contrary to the percentage POH/PONH. As the environment, or area of responsibility, is fully explored, the benefit of maintaining an ε could decrease.

The Boltzmann implementation leads toward a significantly lower overall correctness than the other exploration strategies as depicted in Table 10. This is most likely a product of an unnecessarily high temperature for this particular employment. As the temperature approaches zero, the results should mimic greedy strategies more closely. The higher the temperature the more likely the recommended actions are to be equally probable (Tokic, 2010). Therefore, the recommended action is not necessarily what has the highest reward value, and the system does not get rewarded as frequently, which would alter the upwards progression seen in other runs.

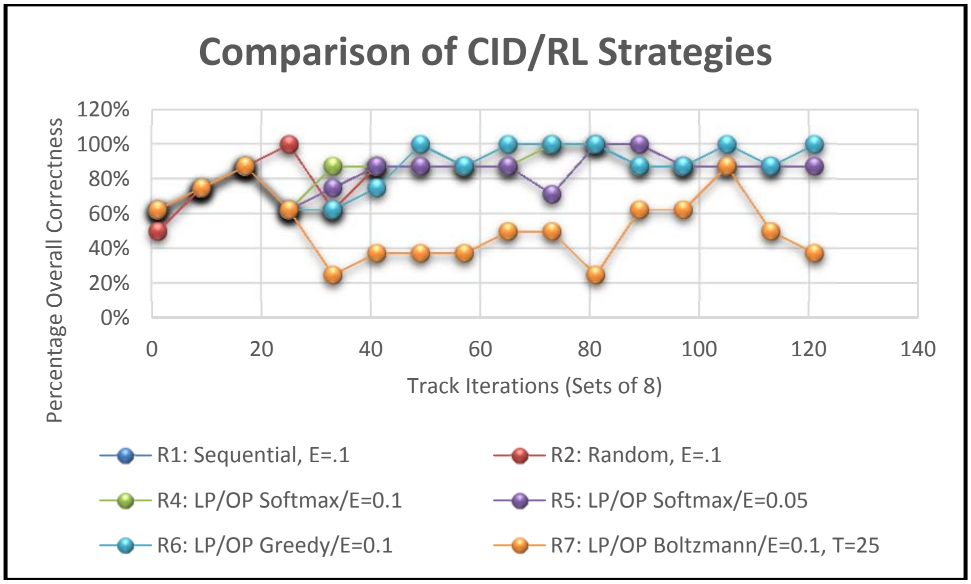
B. HYPOTHESES ANALYSIS

The methodology involved in the analysis primarily depended on comparison of pre-learning metrics to post-learning metrics, overall correctness of evaluations. What performance did the Soar CID application exhibit prior to RL and how did that compare to when RL was enabled?

While it is possible to achieve a relatively stable overall correctness from the beginning by modifying the POH and PONH values to reflect proportionate rewards values based on expected CID metrics, the usage of arbitrary numbers as initial reward values proves that learning has occurred. Baseline non-learning overall correctness was 62.5%, as shown there were multiple configurations of Soar CID parameters that increased the overall correctness. All but one Run of RL implementation showed an improvement over a baseline non-RL sample. The variety of settings and methods available in Soar makes this a powerful tool, but it is imperative to pair the correct parameter settings with the task.

The improvement of the separate RL parameter settings is shown in Figure 7. The outlier that does not improve within the same sample is the Boltzmann configuration in Run 7 (Appendix C). As discussed earlier, this may be due to an inappropriately high temperature setting; further testing should be done to confirm the effect on performance of a lower temperature. As the temperature decreases the Boltzmann algorithm should act more and more like greedy method with a low epsilon. It is possible that a Boltzmann strategy could be useful in this context but this research was not able to thoroughly explore it to ultimately verify it as an acceptable RL strategy for CID.

Figure 7. CID Learning Comparison



Comparison analysis of Runs. Run 3 (LP) not pictured. Run 4–7(OP) first four points are representative of Run 3 (LP). Shows increase in overall correctness for the majority of the parameter selections.

Statistical analysis of the overall correctness will include both the LP and OP. The one-tail *p*-value for non-learning sample to combined LP and OP Run 4 is $p = 0.0027$. Run 6 had the highest overall LP/OP due to the amount of sampling (100); $p = 0.0006$. We reject the null hypothesis since the p-values were less than the alpha value of 0.05 with the exception of Run 7, p=.1206), which was most likely due to an inflated temperature value. Further testing should be completed to explore the effects of lower temperature values on the data. The integration of RL into a rudimentary CID problem was successful. The implementation of a RL/CID system succeeded in a simplistic mimicry of the operator. While the overall correctness was not 100% the improvement displayed from a baseline system to a “learned” system shows that a CID system based on RL is feasible.

V. CONCLUSIONS

A. SUMMARIZATION OF RESULTS

The research question posed in Chapter I was if Reinforcement Learning (RL) can be used effectively for the process of Combat Identification (CID). After developing a basic CID decision-making language, the data developed through the Soar CID Application proved that there is an increase in overall accuracy when RL functions are used. In the continuous Runs (Tables 6 and 7) from untrained to trained, the improvement was small but present, an average improvement from 62.5% to 75.0%. The segregation of phases, to reflect an untrained system LP (learning phase) and a trained system OP (operational phase) were instrumental in proving marked improvement of the system and a reflection of traditional RL performance (Figure 3).

Although the original reward value assignments were not based on any relevant information, the feedback of the operator correctly altered the probability of hostility (POH) and the probability of non-hostility (PONH) to reflect the ground-truth classification of the tracks at a best overall correctness of 91.89%.

While the data did display an increase in overall correctness, the parameter modification for data analysis did not lead to any dramatic epiphanies. The sample size and limited variation of tracks, while an ultimately significant increase, limits the conclusion that one learning method, exploration policy, and learning rate is inherently better than another. RL can be used in conjunction with CID, but there is no definitive combination of parameters that can be identified based on the data.

B. RECOMMENDATION FOR FUTURE RESEARCH

The information gathered in this thesis just scratched the surface of possibilities available to the tools: Soar and RL. At this basic level, proof of concept has been established, but the next steps should verify the results with a larger data set and confirm learning with a more dynamic set of CID Rules. In order to continue development of a Soar CID Application, we recommend the following be completed as the research continues:

- Develop a user interface friendlier to virtual track injection.
- Increase the Ruleset to more accurately portray real life CID parameters.
- Continue parameter evaluation for best fit of CID correctness (i.e. Learning Policy, Rate, and Exploration Strategy).
- Construct track memory for the Soar CID Application.
- Develop automated interface for systems' inputs into Soar CID Application.
- Establish doctrine and policy for integration aboard real world systems.

1. Increase Scale and Complexity

The primary limitations of this research are complexity and scale, as discussed in previous chapters of this thesis. Without a fully vetted and robust ROE and complementary CID matrix it is impossible to fully understand the benefits and uses of SOAR as a decision aid to the TAO/MC in an operational environment. By increasing the CID matrix the variation of tracks also increases, allowing for more rules and more of a sample pool. This will be imperative to test in future research. Can a Soar CID application keep up with a dynamic number track varieties?

Additionally, the basic rules in this research were limited to one or the other, "non-hostile" or "hostile." As the complexity increases, consideration should be given to evaluating other classifications of tracks within the Soar CID application environment and rules, such as developing a variation on "non-hostile" rule for "neutral." An additional possibility is to develop a scale of hostility based on the POH and PONH values. A neutral track could be a certain value of POH or PONH based on real world parameters.

Translating CID functions from plain language to Soar CID Agent language may not be applicable to all of the possible variables that contribute to CID, but as discussed in Chapter III, it could be used to expand the current model for further testing of robustness. While the "tripped" or "not-tripped" concept proved suitable in this scenario, more complex evaluations based on intelligence CID may not translate as fluidly. As the complexity of the ROE and CID increase the usefulness of a plain language method will be verified or disproved.

2. Program Modifications and Extensions

Soar CID is a basic program that does not take advantage of all of the technology available today. The Soar Cognitive Architecture is a versatile program that can be modified in a multitude of means.

One major limitation of the Soar CID Application as it stands is the lack of track memory. This constraint affects the CID process in a few different manners. When the identification variables for a contact are first established, they may not paint a complete picture of the aircraft. As the aircraft continues to operate more identification features may become apparent. As an example, one of the procedural methods of CID discussed in Chapter II is based verifying a flight profile, Return to Force. Without a comparison of flight data at continuous times (*t0, t1, t2, t3...)*, it may be impossible to accurately identify the profile.

While this research limited Soar to interaction with only one other program, Windows Command Prompt, it is possible to write extensions that integrate Soar with other computer programs, which could aid CID evaluations. For instance, Interrogation Friend or Foe (IFF) is a dynamic tool that can lead to an aircraft identifying itself, Mode S, the return could be verified against a public source or database prior to inject of the state conditions to the decision-making agent. The additional database information may be the solution to supplementing any plain language rule construction as discussed above.

3. Weapons System Integration

Developing an interface that automatically injects the sensor values of tracks is one of the first steps to operational usage. The manual entry of track data limitation in the initial Soar CID program is not conducive to operational usage. Further testing in a virtual environment, should require the same improvement to increase realism and allowable sample size.

Consideration should be given to whether or not a Soar CID application is appropriate to an operational environment, and in which manner is the least intrusive to the warfighter. As the research continues, whether a fully "trained" system should be sent directly to operational usage or trained onsite with the rules of engagement appropriate to the theater of operations in mind. Also, if the system should continue to be "trained" when in operational use, updated offline, or stagnate.

4. Parameter and Value Experimentation

While this research has been conducted using a few variations of parameters of which Soar is capable, further research should continue to explore the possible benefits of one learning type over another. Testing the data against a series of exploration strategies using Q-learning over SARSA should be done first.

In Chapter II, we briefly discussed the learning rate modifications. Although this thesis did not delve into the adjustment of the learning rate, consideration should be given to operational usage. As discussed in the previous section, depending on how the Soar CID application would be used operationally, the system can learn at a lower rate, or not at all, in the OP. The selection should depend on the volatility of the environment and the trust of the CID operators. If there are no circumstances in the operating environment with which the RL system does not deftly deal, then there is no reason to leave the learning rate relatively high.

Chapter II briefly discussed parameters and features available in RL and through Soar, while the experimentation limited characteristics of RL based on scale it would be beneficial for future research to thoroughly vet all of the functions for best application to real world situations. This should be done more thoroughly with a more complex CID Ruleset prior to further implementation.

APPENDIX A. RUN 5. OPERATIONAL PHASE, EPSILON .05

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	3	Soar says: not hostile	Y	N	45.8%	54.2%	
2	1	Soar says: hostile	Y	Y	0.0%	100.0%	
3	7	Soar says: not hostile	N	Y	100.0%	0.0%	
4	7	Soar says: not hostile	N	Y	100.0%	0.0%	
5	7	Soar says: not hostile	N	Y	100.0%	0.0%	
6	2	Soar says: hostile	Y	Y	0.0%	100.0%	
7	4	Soar says: hostile	N	N	52.2%	47.8%	
8	3	Soar says: hostile	Y	Y	0.0%	100.0%	
9	1	Soar says: hostile	Y	Y	0.0%	100.0%	
10	2	Soar says: hostile	Y	Y	0.0%	100.0%	
11	8	Soar says: not hostile	N	Y	63.5%	36.5%	
12	7	Soar says: not hostile	N	Y	96.7%	3.3%	
13	1	Soar says: hostile	Y	Y	0.0%	100.0%	
14	4	Soar says: hostile	N	N	54.0%	46.0%	
15	4	Soar says: not hostile	N	Y	100.0%	0.0%	
16	8	Soar says: not hostile	N	Y	73.1%	26.9%	
17	5	Soar says: not hostile	N	Y	100.0%	0.0%	
18	1	Soar says: hostile	Y	Y	0.0%	100.0%	
19	5	Soar says: not hostile	N	Y	100.0%	0.0%	
20	7	Soar says: not hostile	N	Y	100.0%	0.0%	
21	7	Soar says: not hostile	N	Y	100.0%	0.0%	
22	7	Soar says: not hostile	N	Y	100.0%	0.0%	
23	6	Soar says: not hostile	N	Y	100.0%	0.0%	88.89%
24	2	Soar says: hostile	Y	Y	12.8%	87.2%	
25	8	Soar says: not hostile	N	Y	65.0%	35.0%	
26	3	Soar says: hostile	Y	Y	13.2%	86.8%	
27	6	Soar says: not hostile	N	Y	100.0%	0.0%	
28	6	Soar says: not hostile	N	Y	100.0%	0.0%	
29	6	Soar says: not hostile	N	Y	100.0%	0.0%	
30	1	Soar says: hostile	Y	Y	0.0%	100.0%	
31	8	Soar says: hostile	N	N	53.9%	46.1%	
32	8	Soar says: not hostile	N	Y	100.0%	0.0%	
33	8	Soar says: not hostile	N	Y	100.0%	0.0%	
34	3	Soar says: hostile	Y	Y	18.7%	81.3%	
35	1	Soar says: hostile	Y	Y	0.0%	100.0%	
36	8	Soar says: not hostile	N	Y	94.8%	5.2%	
37	3	Soar says: hostile	Y	Y	19.0%	81.0%	
38	4	Soar says: hostile	N	N	43.3%	56.7%	
39	2	Soar says: hostile	Y	Y	41.1%	58.9%	
40	2	Soar says: hostile	Y	Y	29.6%	70.4%	
41	1	Soar says: hostile	Y	Y	0.0%	100.0%	
42	2	Soar says: hostile	Y	Y	24.6%	75.4%	
43	8	Soar says: hostile	N	N	71.1%	28.9%	
44	7	Soar says: not hostile	N	Y	100.0%	0.0%	
45	4	Soar says: hostile	N	N	58.3%	41.7%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
46	7	Soar says: not hostile	N	Y	100.0%	0.0%	
47	5	Soar says: not hostile	N	Y	100.0%	0.0%	
48	8	Soar says: not hostile	N	Y	100.0%	0.0%	
49	3	Soar says: hostile	Y	Y	42.1%	57.9%	
50	1	Soar says: hostile	Y	Y	0.0%	100.0%	
51	7	Soar says: not hostile	N	Y	100.0%	0.0%	
52	1	Soar says: hostile	Y	Y	0.0%	100.0%	
53	4	Soar says: not hostile	N	Y	91.1%	8.9%	
54	5	Soar says: not hostile	N	Y	100.0%	0.0%	
55	4	Soar says: not hostile	N	Y	94.0%	6.0%	
56	7	Soar says: not hostile	N	Y	100.0%	0.0%	
57	3	Soar says: hostile	Y	Y	35.6%	64.4%	
58	4	Soar says: not hostile	N	Y	85.2%	14.8%	
59	3	Soar says: hostile	Y	Y	32.7%	67.3%	
60	7	Soar says: not hostile	N	Y	100.0%	0.0%	
61	1	Soar says: hostile	Y	Y	0.0%	100.0%	
62	8	Soar says: not hostile	N	Y	83.0%	17.0%	
63	5	Soar says: not hostile	N	Y	100.0%	0.0%	
64	8	Soar says: not hostile	N	Y	83.6%	16.4%	
65	5	Soar says: not hostile	N	Y	100.0%	0.0%	
66	2	Soar says: not hostile	Y	N	43.2%	56.8%	
67	2	Soar says: hostile	Y	Y	0.0%	100.0%	
68	1	Soar says: hostile	Y	Y	0.0%	100.0%	
69	4	Soar says: not hostile	N	Y	69.1%	30.9%	
70	4	Soar says: not hostile	N	Y	76.4%	23.6%	
71	2	Soar says: hostile	Y	Y	7.5%	92.5%	
72	2	Soar says: hostile	Y	Y	6.4%	93.6%	
73	7	Soar says: not hostile	N	Y	100.0%	0.0%	
74	3	Soar says: hostile	Y	Y	35.1%	64.9%	
75	4	Soar says: hostile	N	N	68.9%	31.1%	
76	5	Soar says: not hostile	N	Y	100.0%	0.0%	
77	5	Soar says: not hostile	N	Y	100.0%	0.0%	
78	8	Soar says: not hostile	N	Y	61.5%	38.5%	
79	4	Soar says: not hostile	N	Y	100.0%	0.0%	
80	5	Soar says: not hostile	N	Y	100.0%	0.0%	
81	8	Soar says: hostile	N	N	64.5%	35.5%	
82	4	Soar says: not hostile	N	Y	100.0%	0.0%	
83	2	Soar says: hostile	Y	Y	28.4%	71.6%	
84	1	Soar says: hostile	Y	Y	0.0%	100.0%	
85	7	Soar says: not hostile	N	Y	100.0%	0.0%	
86	7	Soar says: not hostile	N	Y	100.0%	0.0%	
87	3	Soar says: not hostile	Y	N	43.4%	56.6%	
88	5	Soar says: not hostile	N	Y	100.0%	0.0%	
89	5	Soar says: not hostile	N	Y	100.0%	0.0%	
90	5	Soar says: not hostile	N	Y	100.0%	0.0%	
91	2	Soar says: hostile	Y	Y	19.3%	80.7%	
92	6	Soar says: not hostile	N	Y	100.0%	0.0%	
93	8	Soar says: not hostile	N	Y	79.7%	20.3%	
94	8	Soar says: not hostile	N	Y	84.7%	15.3%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
95	1	Soar says: hostile	Y	Y	0.0%	100.0%	
96	8	Soar says: not hostile	N	Y	87.0%	13.0%	
97	2	Soar says: hostile	Y	Y	28.9%	71.1%	
98	3	Soar says: hostile	Y	Y	26.0%	74.0%	
99	2	Soar says: hostile	Y	Y	26.7%	73.3%	

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. RUN 6. OPERATIONAL PHASE, GREEDY

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	4	Soar says: hostile	N	N	95.0%	5.0%	62.5%
2	8	Soar says: hostile	N	N	95.0%	5.0%	
3	8	Soar says: not hostile	N	Y	95.0%	5.0%	
4	4	Soar says: not hostile	N	Y	95.0%	5.0%	
5	6	Soar says: not hostile	N	Y	95.0%	5.0%	
6	2	Soar says: not hostile	Y	N	95.0%	5.0%	
7	5	Soar says: not hostile	N	Y	95.0%	5.0%	
8	4	Soar says: not hostile	N	Y	95.0%	5.0%	
9	2	Soar says: hostile	Y	Y	5.0%	95.0%	75.0%
10	5	Soar says: not hostile	N	Y	95.0%	5.0%	
11	4	Soar says: not hostile	N	Y	95.0%	5.0%	
12	2	Soar says: not hostile	Y	N	5.0%	95.0%	
13	8	Soar says: not hostile	N	Y	95.0%	5.0%	
14	6	Soar says: not hostile	N	Y	95.0%	5.0%	
15	3	Soar says: not hostile	Y	N	95.0%	5.0%	
16	1	Soar says: hostile	Y	Y	5.0%	95.0%	
17	5	Soar says: not hostile	N	Y	95.0%	5.0%	100.0%
18	3	Soar says: hostile	Y	Y	5.0%	95.0%	
19	4	Soar says: not hostile	N	Y	95.0%	5.0%	
20	1	Soar says: hostile	Y	Y	5.0%	95.0%	
21	6	Soar says: not hostile	N	Y	95.0%	5.0%	
22	1	Soar says: hostile	Y	Y	5.0%	95.0%	
23	1	Soar says: hostile	Y	Y	5.0%	95.0%	
24	4	Soar says: not hostile	N	Y	95.0%	5.0%	
25	6	Soar says: hostile	N	N	95.0%	5.0%	87.5%
26	3	Soar says: hostile	Y	Y	5.0%	95.0%	
27	4	Soar says: not hostile	N	Y	95.0%	5.0%	
28	3	Soar says: hostile	Y	Y	5.0%	95.0%	
29	4	Soar says: not hostile	N	Y	95.0%	5.0%	
30	2	Soar says: hostile	Y	Y	5.0%	95.0%	
31	8	Soar says: not hostile	N	Y	95.0%	5.0%	
32	7	Soar says: not hostile	N	Y	95.0%	5.0%	
33	2	Soar says: hostile	Y	Y	5.0%	95.0%	100.0%
34	8	Soar says: not hostile	N	Y	95.0%	5.0%	
35	2	Soar says: hostile	Y	Y	5.0%	95.0%	
36	7	Soar says: not hostile	N	Y	95.0%	5.0%	
37	1	Soar says: hostile	Y	Y	5.0%	95.0%	
38	3	Soar says: hostile	Y	Y	5.0%	95.0%	
39	1	Soar says: hostile	Y	Y	5.0%	95.0%	
40	8	Soar says: not hostile	N	Y	95.0%	5.0%	
41	4	Soar says: not hostile	N	Y	95.0%	5.0%	100.0%
42	2	Soar says: hostile	Y	Y	5.0%	95.0%	
43	5	Soar says: not hostile	N	Y	95.0%	5.0%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
44	5	Soar says: not hostile	N	Y	95.0%	5.0%	
45	8	Soar says: not hostile	N	Y	95.0%	5.0%	
46	2	Soar says: hostile	Y	Y	5.0%	95.0%	
47	2	Soar says: hostile	Y	Y	5.0%	95.0%	
48	3	Soar says: hostile	Y	Y	5.0%	95.0%	
49	7	Soar says: not hostile	N	Y	95.0%	5.0%	100.0%
50	7	Soar says: not hostile	N	Y	95.0%	5.0%	
51	2	Soar says: hostile	Y	Y	5.0%	95.0%	
52	7	Soar says: not hostile	N	Y	95.0%	5.0%	
53	1	Soar says: hostile	Y	Y	5.0%	95.0%	
54	3	Soar says: hostile	Y	Y	5.0%	95.0%	
55	1	Soar says: hostile	Y	Y	5.0%	95.0%	
56	6	Soar says: not hostile	N	Y	95.0%	5.0%	
57	5	Soar says: not hostile	N	Y	95.0%	5.0%	87.5%
58	3	Soar says: hostile	Y	Y	5.0%	95.0%	
59	1	Soar says: hostile	Y	Y	5.0%	95.0%	
60	6	Soar says: not hostile	N	Y	95.0%	5.0%	
61	3	Soar says: hostile	Y	Y	5.0%	95.0%	
62	3	Soar says: hostile	Y	Y	5.0%	95.0%	
63	4	Soar says: hostile	N	N	5.0%	95.0%	
64	5	Soar says: not hostile	N	Y	95.0%	5.0%	
65	6	Soar says: not hostile	N	Y	95.0%	5.0%	87.5%
66	5	Soar says: not hostile	N	Y	95.0%	5.0%	
67	8	Soar says: hostile	N	N	5.0%	95.0%	
68	5	Soar says: not hostile	N	Y	95.0%	5.0%	
69	7	Soar says: not hostile	N	Y	95.0%	5.0%	
70	8	Soar says: not hostile	N	Y	95.0%	5.0%	
71	5	Soar says: not hostile	N	Y	95.0%	5.0%	
72	2	Soar says: hostile	Y	Y	5.0%	95.0%	
73	1	Soar says: hostile	Y	Y	5.0%	95.0%	100.0%
74	7	Soar says: not hostile	N	Y	95.0%	5.0%	
75	1	Soar says: hostile	Y	Y	5.0%	95.0%	
76	4	Soar says: not hostile	N	Y	95.0%	5.0%	
77	1	Soar says: hostile	Y	Y	5.0%	95.0%	
78	1	Soar says: hostile	Y	Y	5.0%	95.0%	
79	3	Soar says: hostile	Y	Y	5.0%	95.0%	
80	7	Soar says: not hostile	N	Y	95.0%	5.0%	
81	1	Soar says: hostile	Y	Y	5.0%	95.0%	87.5%
82	6	Soar says: not hostile	N	Y	95.0%	5.0%	
83	4	Soar says: not hostile	N	Y	95.0%	5.0%	
84	4	Soar says: hostile	N	N	95.0%	5.0%	
85	3	Soar says: hostile	Y	Y	5.0%	95.0%	
86	3	Soar says: hostile	Y	Y	5.0%	95.0%	
87	3	Soar says: hostile	Y	Y	5.0%	95.0%	
88	5	Soar says: not hostile	N	Y	95.0%	5.0%	
89	2	Soar says: hostile	Y	Y	5.0%	95.0%	100.0%
90	8	Soar says: not hostile	N	Y	95.0%	5.0%	
91	1	Soar says: hostile	Y	Y	5.0%	95.0%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
92	7	Soar says: not hostile	N	Y	95.0%	5.0%	
93	5	Soar says: not hostile	N	Y	95.0%	5.0%	
94	3	Soar says: hostile	Y	Y	5.0%	95.0%	
95	6	Soar says: not hostile	N	Y	95.0%	5.0%	
96	4	Soar says: not hostile	N	Y	95.0%	5.0%	
97	2	Soar says: hostile	Y	Y	5.0%	95.0%	
98	2	Soar says: hostile	Y	Y	5.0%	95.0%	100.0%
99	2	Soar says: hostile	Y	Y	5.0%	95.0%	

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. RUN 7. BOLTZMANN

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
1	3	Soar says: not hostile	Y	N	50.0%	50.0%	
2	7	Soar says: hostile	N	N	51.1%	48.9%	
3	2	Soar says: not hostile	Y	N	50.0%	50.0%	
4	6	Soar says: hostile	N	N	51.7%	48.3%	
5	3	Soar says: not hostile	Y	N	49.6%	50.4%	
6	7	Soar says: not hostile	N	Y	51.1%	48.9%	
7	3	Soar says: not hostile	Y	N	49.4%	50.6%	
8	1	Soar says: hostile	Y	Y	48.7%	51.3%	
9	2	Soar says: not hostile	Y	N	49.7%	50.3%	
10	6	Soar says: not hostile	N	Y	51.6%	48.4%	
11	8	Soar says: not hostile	N	Y	50.2%	49.8%	
12	5	Soar says: hostile	N	Y	50.7%	49.3%	
13	3	Soar says: not hostile	Y	N	49.5%	50.5%	
14	8	Soar says: hostile	N	N	50.5%	49.5%	
15	7	Soar says: hostile	N	N	51.3%	48.7%	
16	3	Soar says: not hostile	Y	N	49.4%	50.6%	
17	4	Soar says: not hostile	N	Y	50.0%	50.0%	
18	6	Soar says: hostile	N	N	52.4%	47.6%	
19	8	Soar says: hostile	N	N	50.6%	49.4%	
20	8	Soar says: not hostile	N	Y	50.8%	49.2%	
21	3	Soar says: not hostile	Y	N	49.5%	50.5%	
22	7	Soar says: not hostile	N	Y	51.8%	48.2%	48.0%
23	1	Soar says: not hostile	Y	N	48.6%	51.4%	
24	7	Soar says: hostile	N	N	51.8%	48.2%	
25	5	Soar says: hostile	N	N	51.1%	48.9%	
26	1	Soar says: hostile	Y	Y	48.6%	51.4%	
27	2	Soar says: not hostile	Y	N	50.1%	49.9%	
28	1	Soar says: hostile	Y	Y	48.3%	51.7%	
29	1	Soar says: not hostile	Y	N	48.2%	51.8%	
30	4	Soar says: hostile	N	N	49.9%	50.1%	
31	8	Soar says: hostile	N	N	50.9%	49.1%	
32	1	Soar says: hostile	Y	Y	48.4%	51.6%	
33	1	Soar says: hostile	Y	Y	48.3%	51.7%	
34	2	Soar says: hostile	Y	Y	49.8%	50.2%	
35	2	Soar says: not hostile	Y	N	49.5%	50.5%	
36	5	Soar says: not hostile	N	Y	51.2%	48.8%	
37	8	Soar says: hostile	N	N	50.8%	49.2%	
38	6	Soar says: hostile	N	N	52.7%	47.3%	
39	4	Soar says: hostile	N	N	49.8%	50.2%	
40	4	Soar says: not hostile	N	Y	50.1%	49.9%	
41	3	Soar says: not hostile	Y	N	49.7%	50.3%	
42	1	Soar says: hostile	Y	Y	48.3%	51.7%	
43	5	Soar says: not hostile	N	Y	51.5%	48.5%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
44	2	Soar says: hostile	Y	Y	49.4%	50.6%	
45	2	Soar says: not hostile	Y	N	49.2%	50.8%	
46	7	Soar says: hostile	N	N	51.1%	48.9%	
47	1	Soar says: not hostile	Y	N	48.1%	51.9%	
48	6	Soar says: not hostile	N	Y	52.7%	47.3%	
49	1	Soar says: hostile	Y	Y	48.1%	51.9%	
50	8	Soar says: hostile	N	N	50.5%	49.5%	
51	2	Soar says: not hostile	Y	N	49.1%	50.9%	
52	8	Soar says: hostile	N	N	50.6%	49.4%	
53	8	Soar says: hostile	N	N	50.7%	49.3%	
54	6	Soar says: hostile	N	N	52.8%	47.2%	
55	7	Soar says: not hostile	N	Y	51.1%	48.9%	
56	5	Soar says: hostile	N	N	51.6%	48.4%	
57	7	Soar says: not hostile	N	Y	51.2%	48.8%	
58	6	Soar says: hostile	N	N	52.8%	47.2%	
59	5	Soar says: hostile	N	N	51.6%	48.4%	
60	6	Soar says: not hostile	N	Y	52.7%	47.3%	
61	6	Soar says: not hostile	N	Y	52.6%	47.4%	
62	5	Soar says: not hostile	N	Y	51.6%	48.4%	
63	3	Soar says: not hostile	Y	N	49.6%	50.4%	
64	7	Soar says: not hostile	N	Y	51.2%	48.8%	
65	1	Soar says: not hostile	Y	N	48.1%	51.9%	
66	2	Soar says: hostile	Y	Y	49.2%	50.8%	
67	2	Soar says: not hostile	Y	N	48.9%	51.1%	
68	6	Soar says: not hostile	N	Y	52.4%	47.6%	
69	1	Soar says: hostile	Y	Y	47.9%	52.1%	
70	1	Soar says: not hostile	Y	N	47.9%	52.1%	
71	4	Soar says: not hostile	N	Y	50.1%	49.9%	
72	8	Soar says: not hostile	N	Y	50.3%	49.7%	
73	7	Soar says: not hostile	N	Y	51.3%	48.7%	
74	6	Soar says: hostile	N	N	52.8%	47.2%	
75	6	Soar says: not hostile	N	Y	52.7%	47.3%	
76	3	Soar says: hostile	Y	Y	49.6%	50.4%	
77	1	Soar says: hostile	Y	Y	48.0%	52.0%	
78	2	Soar says: hostile	Y	Y	49.0%	51.0%	
79	2	Soar says: hostile	Y	Y	48.9%	51.1%	
80	2	Soar says: hostile	Y	Y	48.8%	51.2%	
81	5	Soar says: not hostile	N	Y	51.4%	48.6%	
82	1	Soar says: hostile	Y	Y	47.8%	52.2%	
83	2	Soar says: not hostile	Y	N	48.7%	51.3%	
84	8	Soar says: not hostile	N	Y	50.1%	49.9%	
85	1	Soar says: hostile	Y	Y	47.8%	52.2%	
86	3	Soar says: not hostile	Y	N	49.5%	50.5%	
87	2	Soar says: not hostile	Y	N	48.8%	51.2%	
88	2	Soar says: not hostile	Y	N	48.6%	51.4%	
89	4	Soar says: not hostile	N	Y	49.8%	50.2%	
90	4	Soar says: hostile	N	N	50.0%	50.0%	

LINE	TRACK #	SOAR RECOMMENDATION	HOSTILE? (Y/N)	Correct?	SOAR % Non-HOSTILITY	SOAR % HOSTILITY	CORRECTNESS
91	7	Soar says: hostile	N	N	51.0%	49.0%	
92	6	Soar says: hostile	N	N	52.7%	47.3%	
93	7	Soar says: hostile	N	N	51.2%	48.8%	
94	1	Soar says: hostile	Y	Y	48.2%	51.8%	
95	8	Soar says: not hostile	N	Y	50.3%	49.7%	
96	5	Soar says: hostile	N	N	52.1%	47.9%	
97	3	Soar says: hostile	Y	Y	49.7%	50.3%	
98	2	Soar says: hostile	Y	Y	49.1%	50.9%	
99	3	Soar says: hostile	Y	Y	49.4%	50.6%	
100	8	Soar says: hostile	N	N	50.3%	49.7%	

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Alt, J. K. (2012). *Learning from noisy and delayed rewards the value of reinforcement learning to defense modeling and simulation* (Doctoral dissertation). Retrieved from Calhoun http:/hdl.handle.net

Bryant, D. J. (2009). *Combat identifiation (15au): Project summary and closeout report*. Toronto, Canada: Defense R&D Canada.

Chief of Naval Operations (CNO). (2012, Jul. 16). *Standard organization and regulations of the U.S. Navy* (OPNAVINST 3120.32D). Washington, DC: Author.

Dearden, R. W. (2000). *Learning and planning in structured words* (Doctoral thesis). https://open.library.ubc.ca/cIRcle/collections/ubctheses/831/items/1.0051620

Department of Defense and Joint Chiefs of Staff (DOD and JCS). (1996). *Joint warfighting science and technology plan.* Washington, DC: Author.

Department of the Navy Office of the Chief of Naval Operations(CNO). (2014). (U) *DRAFT: Maritime air and missile defense planning*, (NTTP 3-01.11). Washinton, DC: Author. This document is classified SECRET.

Department of the Navy. (2013). Mark XIIA identification friend or foe (IFF) mode 5. Retrieved from http://www.dote.osd.mil/pub/reports/FY2013/pdf/navy/2013mkxiiaiffmode5.pdf

Dottery, K. A. (1992). *Systematic analysis of complex dynamic systems: the Case of the USS Vincennes* (Master's thesis). Retrieved from http://hdl.handle.net/10945/30594

Eden, T., Knittel, A., & Uffelen, R. (n.d.). Algorithms. Retrieved March 13, 2017, from http://www.cse.unsw.edu.au/~cs9417ml/RL1/algorithms.html

HQ TRADOC. (2002). *Brevity: Multiservice brevity codes.* Fort Monroe, VA: Author.

Joint Staff. (2014, Dec 12). *Joint fire support* (Joint Publication 3-09). Washington, DC: Author.

Laird, J. E. (2012). *The Soar cognitive architecture* . Cambridge, MA: The MIT Press.

Laird, J. E., & Congdon, C. B. (2015). *The Soar user's manual: Version 9.5.0*. Ann Arbor, MI: University of Michigan.

Lewicki, M. S. (2007, May). Reinforcement learning I: Artificial intelligence: Representation and problem solving [PDFSlides]. Retrieved from Carnegie Melon https://www.cs.cmu.edu /af s/cs/academic/class/15381-s07/www/slides/050107 reinforcementLearning1.pdf

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: The MIT Press.

Tokic, M. (2010). Adaptive e-greedy exploration in reinforcement learning based value differences. *Lecture notes in computer science*: *Vol. 6359. Advances in artificial intelligence* (pp. 203-210). doi: 10.1007/978-3-642-16111-7_23

Transponder Requirements. (2006). [Equipment information]. Retrieved from AOPA: https://www.aopa.org/training-and-safety/pic-archive/equipment/transponder-requirements

United States Air Force (USAF). (2014). Dynamic targetting and the tasking process. Retrieved from Curtis E. Lemay Center https://doctrine.af.mil/download.jsp?filename=3-60-D17-Target-Dynamic-Task.pdf

Wang, Y., & Ruhe, G. (2007). The Cognitive process of decision making. *Int'l Journal of Cognitive Informatics of Natural Intelligence, 1*(2), 73–85. Retrieved from http://www.ucalgary.ca/icic/files/icic/67-IJCINI-1205-DecisionMaking.pdf

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California